

Chapter 3

RELATIONAL MODEL

Chapter 3 – Objectives

- ▶ Terminology of relational model.
- ▶ How tables are used to represent data.
- ▶ Connection between mathematical relations and relations in the relational model.
- ▶ Properties of database relations.
- ▶ How to identify candidate, primary, and foreign keys.
- ▶ Meaning of entity integrity and referential integrity.
- ▶ Purpose and advantages of views.

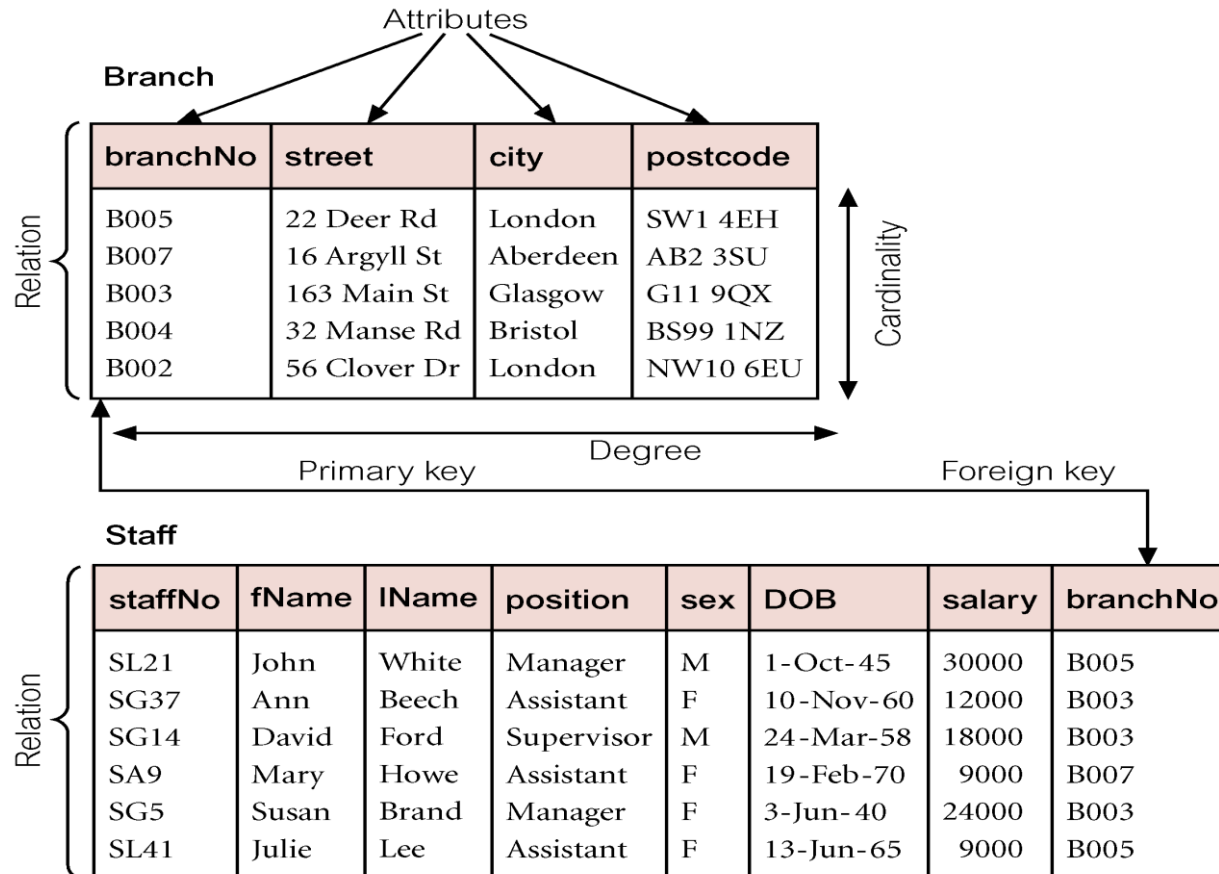
Relational Model Terminology

- ▶ **A relation is a table with columns and rows.**
 - Only applies to logical structure of the database, not the physical structure.
- ▶ **Attribute is a named column of a relation.**
- ▶ **Domain is the set of allowable values for one or more attributes.**

Relational Model Terminology

- ▶ **Tuple is a row of a relation.**
- ▶ **Degree is the number of attributes in a relation.**
- ▶ **Cardinality is the number of tuples in a relation.**
- ▶ **Relational Database is a collection of normalized relations with distinct relation names.**

Instances of Branch and Staff (part) Relations



Examples of Attribute Domains

Attribute	Domain Name	Meaning	Domain Definition
branchNo	BranchNumbers	The set of all possible branch numbers	character: size 4, range B001–B999
street	StreetNames	The set of all street names in Britain	character: size 25
city	CityNames	The set of all city names in Britain	character: size 15
postcode	Postcodes	The set of all postcodes in Britain	character: size 8
sex	Sex	The sex of a person	character: size 1, value M or F
DOB	DatesOfBirth	Possible values of staff birth dates	date, range from 1-Jan-20, format dd-mmm-yy
salary	Salaries	Possible values of staff salaries	monetary: 7 digits, range 6000.00–40000.00

Alternative Terminology for Relational Model

Table 3.1 Alternative terminology for relational model terms.

Formal terms	Alternative 1	Alternative 2
Relation	Table	File
Tuple	Row	Record
Attribute	Column	Field

Database Relations

▶ Relation schema

- Named relation defined by a set of attribute and domain name

▶ Relational database schema

- Set of relation schemas, each with a distinct name.

Properties of Relations

- ▶ Relation name is distinct from all other relation names in relational schema.
- ▶ Each cell of relation contains exactly one atomic (single) value.
- ▶ Each attribute has a distinct name.
- ▶ Values of an attribute are all from the same domain.

Properties of Relations

- ▶ Each tuple is distinct; there are no duplicate tuples.
- ▶ Order of attributes has no significance.
- ▶ Order of tuples has no significance, theoretically.

Relational Keys

▶ Primary Key

- Candidate key selected to identify tuples uniquely within relation.

▶ Alternate Keys

- Candidate keys that are not selected to be primary key.

▶ Foreign Key

- Attribute, or set of attributes, within one relation that matches candidate key of some (possibly same) relation.

Relational Integrity

▶ Null

- Represents value for an attribute that is currently unknown or not applicable for tuple.
- Deals with incomplete or exceptional data.
- Represents the absence of a value and is not the same as zero or spaces, which are values.

Relational Integrity

▶ Entity Integrity

- In a base relation, no attribute of a primary key can be null.

▶ Referential Integrity

- If foreign key exists in a relation, either foreign key value must match a candidate key value of some tuple in its home relation or foreign key value must be wholly null.

Relational Integrity

- ▶ **Enterprise Constraints**
 - Additional rules specified by users or database administrators.

Views

▶ Base Relation

- Named relation corresponding to an entity in conceptual schema, whose tuples are physically stored in database.

▶ View

- Dynamic result of one or more relational operations operating on base relations to produce another relation.

Views

- ▶ A virtual relation that does not necessarily actually exist in the database but is produced upon request, at time of request.
- ▶ Contents of a view are defined as a query on one or more base relations.
- ▶ Views are dynamic, meaning that changes made to base relations that affect view attributes are immediately reflected in the view.

Purpose of Views

- ▶ Provides powerful and flexible security mechanism by hiding parts of database from certain users.
- ▶ Permits users to access data in a customized way, so that same data can be seen by different users in different ways, at same time.
- ▶ Can simplify complex operations on base relations.

Updating Views

- ▶ All updates to a base relation should be immediately reflected in all views that reference that base relation.
- ▶ If view is updated, underlying base relation should reflect change.

Updating Views

- ▶ There are restrictions on types of modifications that can be made through views:
 - Updates are allowed if query involves a single base relation and contains a candidate key of base relation.
 - Updates are not allowed involving multiple base relations.
 - Updates are not allowed involving aggregation or grouping operations.